

RECEIVED
CENTRAL FAX CENTER

JUL 25 2006

REMARKS/ARGUMENTS

Pending claims 1-10 and 17-30 stand rejected under 35 U.S.C. §102(e) over U.S. Patent No. 6,510,508 (Zuraski). Applicants respectfully traverse the rejection, and respectfully request reconsideration of the same. As to claim 1, Zuraski nowhere teaches either: (1) invalidating an entry of a filter if an update to the entry occurs during a context; or (2) flushing a portion of a pipeline resource coupled to the filter corresponding to an address space including the entry. In this regard, the Office Action refers to col. 9, lns. 48-61 and col. 11, lns. 14-20 of Zuraski to contend presence of the claimed invalidating of an entry. However, nowhere do these or any other portions of Zuraski anywhere teach that an entry in flush filter 40 is invalidated if an update to the entry occurs during a context. Instead, both of these portions only teach that a flush filter 40 asserts an Invalidate signal out to a translation lookaside buffer (TLB) 39. Nowhere do these or any other portions of Zuraski teach that such Invalidate signal invalidates an entry of the filter itself. Certainly, it nowhere teaches invalidating such an entry if an update to the entry occurs during a context.

Furthermore, nowhere does Zuraski anywhere teach flushing a portion of a pipeline resource corresponding to an address space including such an entry. In this regard, the Office Action refers to col. 13, lns. 3-11 of Zuraski. However, neither this nor any other portion of Zuraski teaches such flushing of a portion of a resource. Instead, all that Zuraski teaches is that upon the occurrence of certain events, flush filter 40 causes a flush of a TLB 39 *in its entirety*. That is, Zuraski, instead teaches:

In the embodiment shown, a flush of TLB may occur when filter circuit 403 asserts an Invalidate signal, *thereby invalidating all entries currently stored in TLB 39*.

Zuraski, col. 11, lns. 14-17 (emphasis added). Clearly, Zuraski contemplates that the *entire* TLB is flushed.

Nor does the passage of Zuraski cited in col. 13 anywhere teach flushing of a portion of a pipeline resource. Instead, this portion of Zuraski merely teaches:

As such, it is possible that some address translations loaded into the TLB from page table B are no longer valid. Consequently, *the TLB flush filter may allow a TLB flush* (shown here as a filtered flush) to occur following the next context switch.

Zuraski, col. 13, ins. 7-11 (emphasis added). Clearly, this does not teach the flushing of a portion; instead it is a full TLB flush. For at least these reasons, Zuraski does not teach flushing of only a portion of a pipeline resource, and certainly not a portion corresponding to an address space including an invalidated and updated entry of a filter. Accordingly, claim 1 and its dependent claims are patentable.

Dependent claim 4 is further patentable, as Zuraski nowhere teaches comparing an address obtained from an external snoop to entries in the filter to determine if an update has occurred. In this regard, the Office Action refers to col. 10, ins. 1-17. All this portion of Zuraski teaches is that a flush filter 40 may receive a snoop request signal and a snoop address. However, Zuraski only teaches that this information is used "for searches of a region table." Nowhere does Zuraski teach that an entry in a flush filter is invalidated based on a comparison of an address obtained from an external snoop.

Dependent claim 5 further recites flushing the portion of the pipeline resource via microcode. Zuraski does not teach that its flush operation is performed by microcode, contrary to the Office Action's contention. Instead, Zuraski merely teaches that an Invalidate signal is asserted to allow a flush of TLB 39. Zuraski, col. 9, ins. 53-55. To support the rejection of claim 5, the Office Action instead refers to col. 4, ins. 11-29 of Zuraski. However, all this portion of Zuraski teaches is an instruction cache includes tags as to whether an instruction is executed by invoking a microcode procedure. This portion has no relevance to the operation of the flush filter. As a result, there is no teaching of the claim 5 subject matter, as recited in the claim, and accordingly claim 5 is patentable. *In re Bond*, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990) (every element of claim must be identically shown in a single reference).

As to independent claim 6, for similar reasons Zuraski nowhere teaches flushing a portion of entries of a pipeline resource if one of the portion is updated during a context. For claim 6, the Office Action only refers to the same portions of Zuraski cited above with regard to claim 1. Because Zuraski nowhere teaches that its TLB has only a portion that is flushed if one of its entries is updated during a context, claim 6 and its dependent claims are patentable over Zuraski.

As to independent claim 17, Zuraski nowhere teaches a pipeline resource having multiple address spaces that are each selectively flushable while the others are maintained. Instead, as described above, the TLB of Zuraski is flushed in its entirety. As to the recitation in the Office Action of claim 17 (*see* Final Office Action, p. 3), the Office Action ignores the prior

amendment to claim 17. Because nothing in Zuraski teaches maintaining other address spaces in a pipeline resource while one address space is selectively flushed, claim 17 is patentable. To the extent that the Office Action somehow contends that col. 13, lns. 3-11 meets such maintaining of entries, this contention is respectfully traversed. As discussed above, this portion of Zuraski, as with all others, teaches an entire flush of the TLB 39. Accordingly, claim 17 and its dependent claims are patentable.

With regard to independent claim 21, Zuraski nowhere teaches either dynamic partitioning of a filter into multiple partitions, nor where such partitions each correspond to one of multiple address spaces. In this regard, there is no teaching anywhere in Zuraski of dynamic partitioning. Instead, the cited portion of Zuraski (i.e., col. 10, lns. 27-65) merely sets forth the fixed structure of the flush filter of Zuraski. That is, while the circuitry of the flush filter shown in FIG. 3 of Zuraski and described in col. 10 includes different components, nowhere does Zuraski teach that such components are dynamically partitioned. Instead, it appears the opposite is true: the structures detailed in col. 10 of Zuraski are fixed and thus are not dynamically partitioned.

Further, Zuraski does not teach that these components are partitioned into partitions each corresponding to one of multiple address spaces. Instead, all that the Office Action cites in this regard is that a region table includes a CAM and a RAM to store addresses and tags. Nowhere however does Zuraski teach that such storage is partitioned such that each partition corresponds to only one of multiple address spaces. Accordingly, claim 21 and its dependent claims are patentable. For at least similar reasons, claim 25 and its dependent claims are patentable as Zuraski nowhere teaches the recited dynamic partitioning of a filter of a pipeline resource into multiple partitions each corresponding to one of multiple address spaces.

Dependent claim 23 further recites that each of the multiple partitions of a filter includes a fixed portion, and the filter further includes a dynamic portion. As contended support for this recited subject matter, the Office Action refers to a paragraph in the Summary of Invention (specifically lns. 36-51 of col. 2). Nowhere does this or any other portion of Zuraski anywhere teach that the flush filter 40 includes fixed partitions and a dynamic portion. Accordingly, for this further reason the rejection of claim 23 is overcome.

As to dependent claim 24, Zuraski further nowhere teaches allocating at least part of the dynamic portion to an application that has consumed the fixed portion of its partition. In fact,

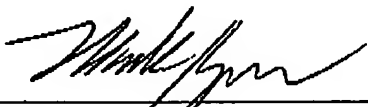
Zuraski appears to teach the opposite. That is, Zuraski teaches that when the region table of the flush filter 40 runs out of available resources an overflow signal is asserted. Furthermore, this overflow signal deactivates the flush filter 40. Zuraski, col. 10, lns. 49-65. Clearly, there is no teaching of allocating a dynamic portion if an application has consumed a fixed portion. Instead, an overflow occurs and the flush filter of Zuraski is simply deactivated. For this further reason claims 24 and 27 are further patentable.

Independent claim 28 is patentable for similar reasons described above with regard to claim 17, as Zuraski nowhere teaches a pipeline resource with multiple address spaces each corresponding to one of multiple contexts, and where each of the address spaces is selectively flushable while the others are maintained. Accordingly, claim 28 and its dependent claims are patentable. This is especially so, as again the Office Action fails to acknowledge the previous amendments made to claim 28.

In view of these remarks, the application is now in condition for allowance and the Examiner's prompt action in accordance therewith is respectfully requested. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 20-1504.

Respectfully submitted,

Date: 7/25/06



Mark J. Rozman
Registration No. 42,117
TROP, PRUNER & HU, P.C.
1616 S. Voss Road, Suite 750
Houston, Texas 77057-2631
(512) 418-9944 [Phone]
(713) 468-8883 [Fax]
Customer No.: 21906